

Android-разработчик. Продвинутый курс

Курс обеспечивает глубокое погружение в промышленную Android-разработку на языке Kotlin, и за 5 месяцев позволяет прокачать свои навыки с уровня Junior до Middle или Senior Android-разработчика.

Длительность курса: 68 академических часов

1 Kotlin

- | | | |
|---|---|---|
| 1 | Kotlin стандартная библиотека. Часть 1 | После этого занятия вы сможете оперировать основными конструкциями языка Kotlin JVM, сможете писать код в kotlin стиле, использовать библиотеку anko. |
| 2 | Kotlin стандартная библиотека. Часть 2 | После этого занятия вы сможете оперировать основными конструкциями языка Kotlin JVM, сможете писать код в kotlin стиле, использовать библиотеку anko. |
| 3 | Kotlin coroutines | После занятия вы сможете использовать корутины и каналы для написания асинхронного кода. |

2 Android Framework

- 1 Advanced Fragment**

Child Fragments, Headless Fragments, Fragments communications, Single Activity Application.
После занятия вы сможете применить полученные знания при проектировании приложения, полностью построенного на Фрагментах.

- 2 Advanced Service**

JobService, BoundService, IPC, AIDL, Handler/Messenger
После этого занятия сможете грамотно выбирать необходимую реализацию Android Service исходя из задачи и версии ОС, научитесь выстраивать коммуникации между сервисами, которые работают в разных процессах.

- 3 Advanced Views**

View Lifecycle, measurement, drawing, SurfaceView, Constraint Layout.
После занятия вы сможете применять знания о процессах отрисовки view при создании собственных имплементаций Android View.

- 4 Advanced Canvas**

Canvas, Bitmap, PorterDuff, Matrix, Path.
После занятия вы сможете использовать возможности Canvas API из Android SDK для рисования сложных оптимизированных view.

- 5 UI Profiling**

GPU profiling, dumpsys, GPS measurement, слои наложения.
После занятия вы сможете использовать различные инструменты для поиска потенциальных узких мест в отрисовке UI и оптимизировать низкопроизводительные решения для быстрой отрисовки ваших приложений.

- | | | |
|----|---------------------------------------|---|
| 6 | Network | Interceptors, CallAdapter.Factory, SSL pinning, WebSocket connection, GSON.
После этого занятия вы сможете применять широкие возможности http клиента Retrofit и десериализатора Gson для построения сетевого уровня в своем приложении. |
| 7 | Security in Android | Android Keystore, Google tink, алгоритмы шифрования в java cipher/android SDK.
После этого занятия вы сможете выбрать оптимальный алгоритм шифрования для вашего случая, сможете организовать безопасное хранилище секретных данных |
| 8 | Proguard | После занятия вы сможете встроить Proguard обфускацию и оптимизацию в свое приложение. |
| 9 | Reverse Engineering | APKtool, jadx.
После этого занятия вы сможете разобрать .apk файл и проверить код на наличие вредоносных действий. |
| 10 | Android studio tips and tricks | После этого занятия вы сможете эффективно использовать возможности Android Studio для профилирования и отладки. |

3 Architecture

- 1 Design patterns**

GoF паттерны, publisher/ subscriber, LRU etc.
После этого занятия вы научитесь применять на практике часто используемые паттерны проектирования из GoF, паттерны многопоточного взаимодействия и синхронизации, кеширования и т.п.

- 2 MV* patterns in Android**

Архитектурные паттерны: MVP/ MVVM/MVI.
После занятия вы сможете подбирать нужный архитектурный проект и собирать проект на основе данной архитектуры.

- 3 Clean Architecture**

После занятия вы сможете выстраивать архитектуру в соответствии с принципами Clean Architecture сформулированными Бобом Мартином.

- 4 Android Architecture Components**

LifecycleOwner, LiveData, Room, Paging, ViewModel.
После занятия вы сможете применять AAC в своих приложениях, адаптировать их под конкретно выбранную архитектуру.

- 5 Dagger 2**

Dagger2, Dagger2 Android Injector.
Вы сможете правильно организовывать DI в приложении, используя Dagger2.

- 6 RxJava 2**

После занятия вы научитесь применять “недооцененные” операторы, писать собственные операторы, узнаете как работают шедулеры.

- 7 Multimodule project**

Dagger2, Gradle modules.
После этого занятия вы научитесь проектировать мультимодульную архитектуру приложения.

8 **Code smells + refactoring**

После этого занятия вы научитесь находить плохие архитектурные решения, строить roadmap и проводить рефакторинг.

4 Testing

- 1 Юнит тесты**
JUnit, Hamcrest matchers.
После этого занятия вы сможете покрывать unit тестами ваши классы.

- 2 Интеграционное тестирование на Android**
Mockito, Robolectric.
После этого занятия вы сможете использовать фреймворки для интеграционного тестирования на Android, реализовывать различные виды тестовых дублеров, создавать shadowклассы.

- 3 Automated UI Testing**
Espresso framework, Spoon, MockWebserver .
После этого занятия вы сможете писать UI тесты на Espresso в изолированной среде и запускать их на нескольких эмуляторах параллельно.

5 Infrastructure and build processes

- 1 Инструменты для проверки качества кода**

Android Lint, findbugs, checkstyle, find-sec-bugs, ktlint, detekt.
После этого занятия вы сможете внедрять и настраивать линтеры/чекстайлы для проверки качества кода и поиска потенциальных ошибок.

- 2 Системы контроля версий, различные виды ветвлений, код ревью**

gitflow, trunk-based development, code review.
По окончании этого занятия вы сможете сравнить различные типы ветвления в git, научитесь эффективно проводить code review своим коллегам.

- 3 Gradle. Часть 1**

Gradle Task, Gradle Plugin, Gradle DSL syntax.
После занятия вы сможете писать свои Gradle Tasks и Gradle Plugin, подключать их в проект и встраивать в пайплайн.

- 4 Gradle. Часть 2**

Gradle Task, Gradle Plugin, Gradle DSL syntax.
После занятия вы сможете писать свои Gradle Tasks и Gradle Plugin, подключать их в проект и встраивать в пайплайн.

- 5 Docker**

Docker, DockerRegistry, Docker Compose, Docker CLI.
После занятия вы научитесь собирать докер образ из Dockerfile, поднимать контейнер из образа, взаимодействовать с контейнером.

6 **Jenkins**

Jenkins, Jenkins pipeline.

После занятия вы научитесь поднимать Jenkins в Docker контейнере и делегировать ему сборку Android проекта.

7 **Distribution / Crash overview**

Fabric, Firebase Crashlytics, AppCentre, Google play.

По окончании этого занятия вы сможете деплоить ваш .apk файл в системы для дистрибуции приложений.

6 Проектная работа

1 **Консультация
по ДЗ и
проектам**

2 **Консультация
по ДЗ и
проектам**

3 **Защита
проектов**